

Problem 2 - Thresholding

```
# Dependencies
import os
import cv2 as cv
from matplotlib import pyplot as plt
import numpy as np

# * User Input for image and choosing the emphasis region
input_image = input("Which image to process [circuit, crack]? : ")
input_processing = int(input("Emphasize (1. Bright) or (2. Dark) regions? (Enter 1 or 2) : "))

# * Reading the image as per user input and displaying it
input_image_path = os.path.join("../ps1-images", input_image)

img = cv.imread(input_image_path + ".png")
if img is None:
    print("Could not open or find the image: ", input_image_path)
    exit(0)
cv.imshow("Original Image", img)

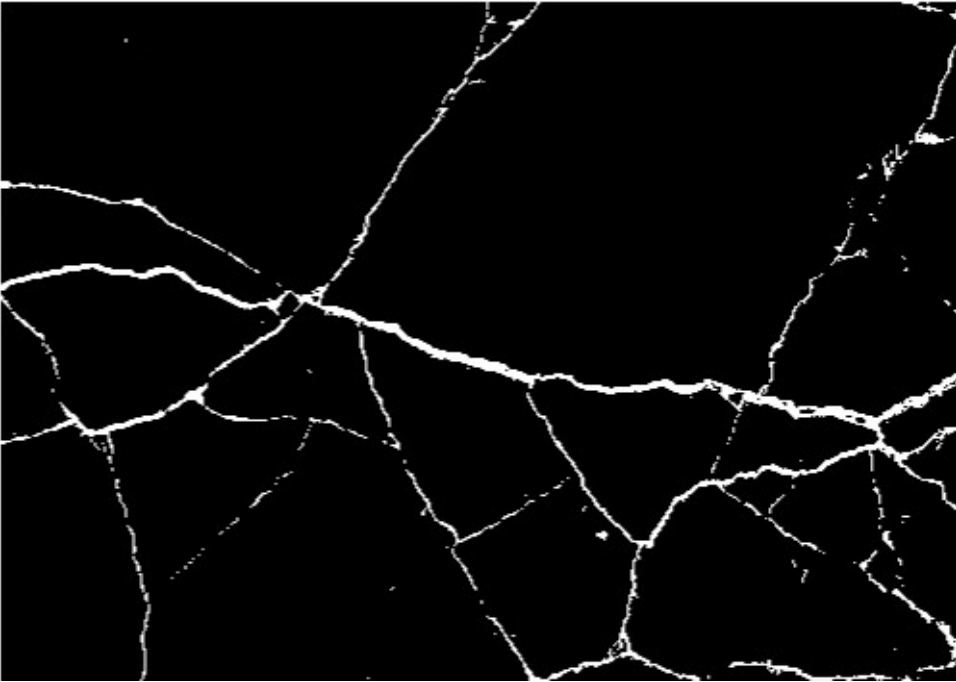
# * Converting to grayscale and displaying it
img_gray = cv.cvtColor(img, cv.COLOR_BGR2GRAY)
cv.imwrite(input_image + "_grayscale.png", img_gray)
cv.imshow("Grayscale Image", img_gray)

# * Thresholding as per emphasis region and displaying it
thresholds = {"crack": 150, "circuit": 80} # Fixed values for global thresholding by observation
if input_processing == 1:
    T, img_thres = cv.threshold(img_gray, thresholds[input_image], 255, cv.THRESH_BINARY)
else:
    T, img_thres = cv.threshold(img_gray, thresholds[input_image], 255, cv.THRESH_BINARY_INV)
cv.imwrite(input_image + "_binary.png", img_thres)
cv.imshow("Binary Image", img_thres)

# * Colouring the emphasis region 'Red' and displaying it
img_red = img
img_red[img_thres == 255] = [0, 0, 255]
cv.imwrite(input_image + "_output.png", img_red)
cv.imshow("Processed Image", img_red)

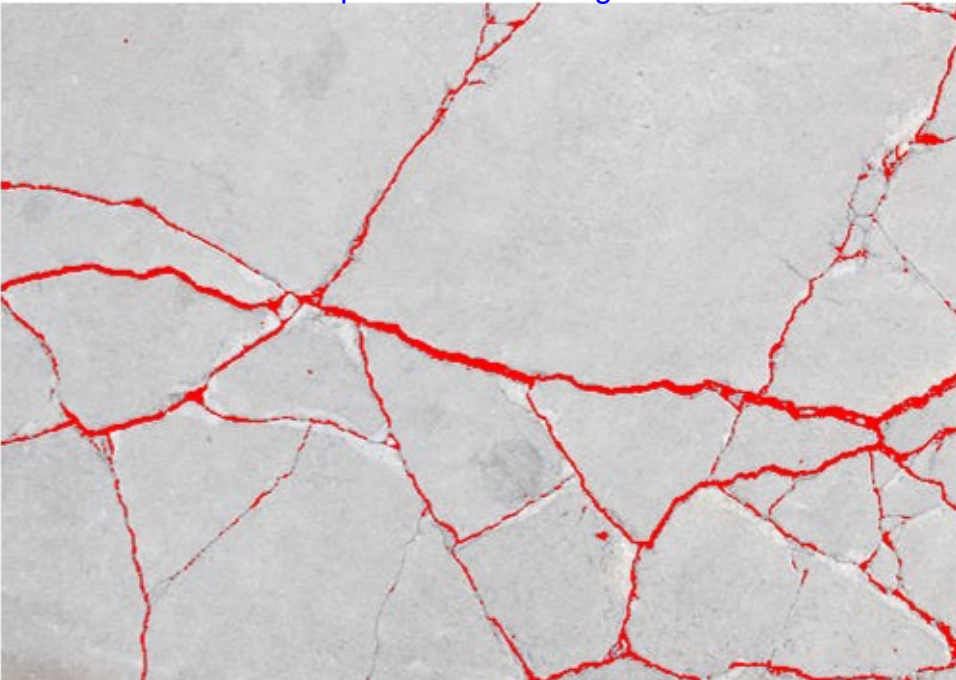
cv.waitKey(10000)
cv.destroyAllWindows()
```

Binary



Crack

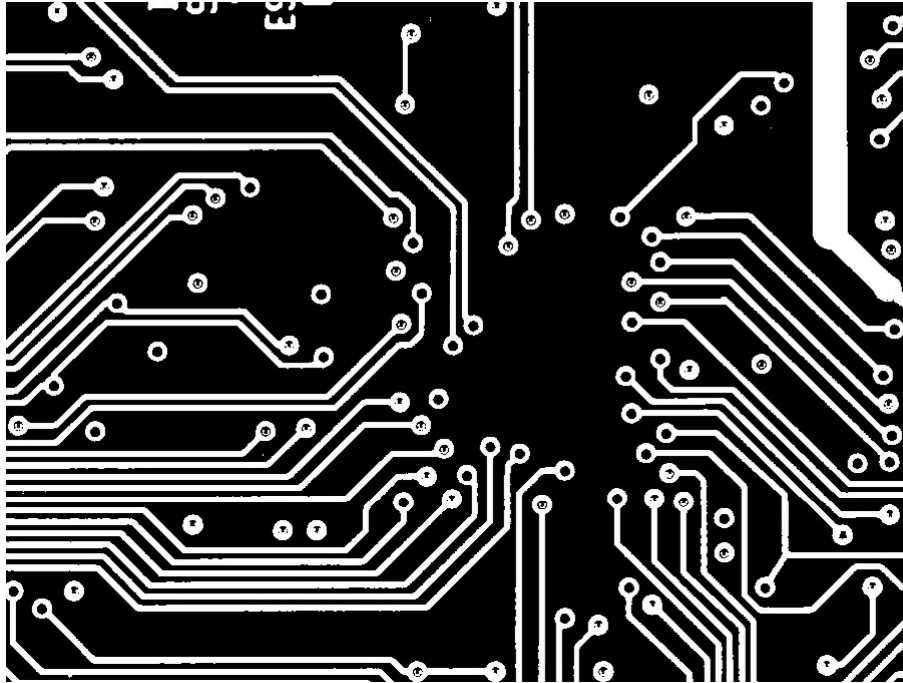
Emphasized Dark Regions



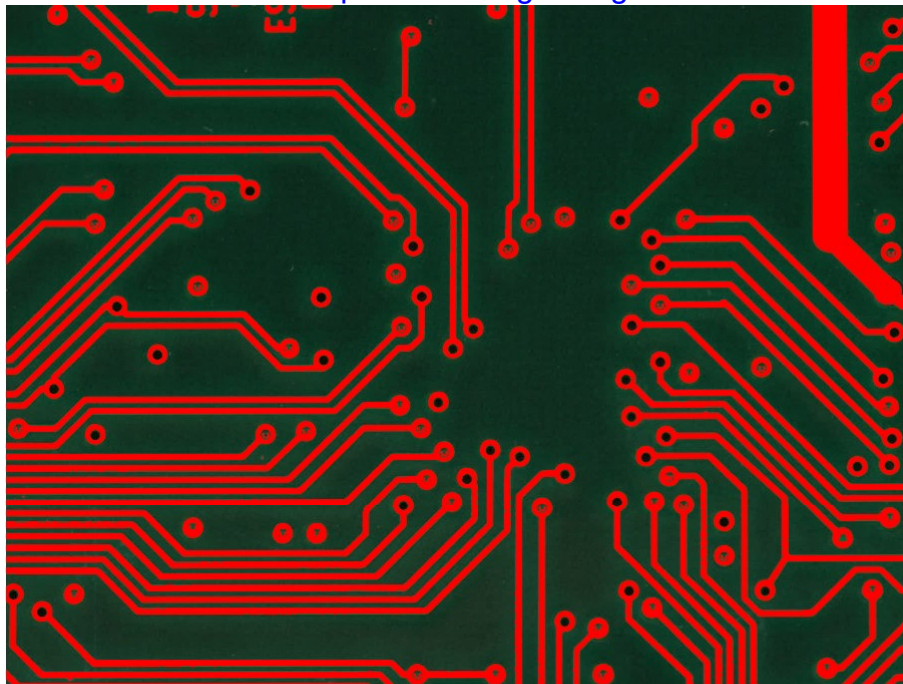
Grayscale



Binary



Emphasized Bright Regions



Circuit

Grayscale

